An Introduction to PatPilr

Raphaël Jauslin

10 décembre 2018

Table des matières

1 Overview

This package is a tool to facilitate the pre-treatment and the treatment of NGS data. The tools are implemented to work on fastq and fasta file. This introduction will, step-by-step, explain how to use the package and which functions you should use in order to obtain your data merged, demultiplexed and cleaned. This will supposed that you have two .fastq R1 and R2 and a barcode file that contains the informations for the demultiplexing step. Firstly, we need to download and install the package PatPilr. You could find the package in the github repository : https://github.com/RJauslin/PatPilr. You should launch the following commands in `R` or `Rstudio` in order to install PatPilr.

```
install.packages("devtools")
devtools::install_github("Rjauslin/PatPilr@master")
```

2 Pre-treatment

This is the first step of the pipeline. The input are the original files from the server they are generally named `xxx_R1.fastq` and `xxx_R2.fastq`. We will explain how the program deal with the main three steps, merging, demultiplexing, and cleaning. During all the process we will supposed that you have put only the two fastq files and the informations needed for the demultiplexing (see Section 2.2).

```
library(PatPilr)

#Linux
pathFolder <- "/home/raphael/Documents/......./working_directory/"
#Windows
pathFolderWindows <- "C:/Users/raphael/......./working_directory/"
```

All the pre-treatment is wrap inside a function that call the different functions. So you only have to check the parameter of the function and click enter.

```
preTreatment(pathFolder,
  m = 10, # min overlap
  M = 100, # max overlap
  x = 0.25, # max mismatch density
  t = 4, # number of threads
  mismatch = FALSE, # allows 1 mismatch in tag if TRUE
  err = 0.01, #
  slide = 50, #
  minlength = 60) # parameters for qual check
```

2.1 Merging

The merging step currently implemented is done by the program FLASH [Magoč and Salzberg, 2011]. We have allows some possible parameters

m The minimum required overlap length between two reads to provide a confident overlap.

M Maximum overlap length expected in approximately 90% of read pairs.

x Maximum allowed ratio between the number of mismatched base pairs and the overlap length.

t Set the number of worker threads.

2.2 Demultiplexing

2.2.1 Simple tag

The demultiplexing step is implemented by the program `PatPil` that is hide in the package. The function calls the tools `D_simple_tag` that could be used from the shell by the following command.

```
./PatPil D_simple_tag -f ./merged.fastq -o ./outputFolder/ -b ./barcodes.txt -mismatch
```

The only thing that you should care is that your barcode.txt file is of the following form.

```
ACGAGTGCGT 01.fq
ACGCTCGACA 02.fq
AGACGCACTC 03.fq
AGCACTGTAG 04.fq
ATCAGACACG 05.fq
ATATCGCGAG 06.fq
CGTGTCTCTA 07.fq
CTCGCGTGTC 08.fq
...
```

More important the separator between the tags and the names of the files should be a tab.

2.2.2 Double tag

```
ACACACAC ForwardTag1
ACGACTCT ForwardTag2
ACGCTAGT ForwardTag3
ACTATCAT ForwardTag4
...
```

```
ACACACAC ReverseTag1
ACGACTCT ReverseTag2
ACGCTAGT ReverseTag3
ACTATCAT ReverseTag4
...
```

```
CAAAATCATAAAGATATTGGDAC GAAATTTCCDGGDTATMGAATGG
```

2.3 Quality check

[Edgar and Flyvbjerg, 2015]

3 Dereplication

Références

[Edgar and Flyvbjerg, 2015] Edgar, R. C. and Flyvbjerg, H. (2015). Error filtering, pair assembly and error correction for next-generation sequencing reads. *Bioinformatics*.

[Magoč and Salzberg, 2011] Magoč, T. and Salzberg, S. L. (2011). FLASH : Fast length adjustment of short reads to improve genome assemblies. *Bioinformatics*.